

Documentation of optimization tool

Description

Fritolay.py is a tool for the choice of SKUs in small format for Southwest region, help regional sales directors better select SKUs.

managerial overhead and reduced inventory efficiency when there are many SKUs. While still maximizing profits (estimated profits – transportation costs).

Using Fritolay.py

As the first step, you need to download and save the Fritolay.py and input files into the same path.

The files needed are

- Estimated margins
estimates of the profit margins of FritoLays for various products.
Our sample input is the file 'estimated_margins.csv'
- sales
all sales of FritoLays products in 2018 in each sales period (each period is 4 weeks). This data records sales in each region to each retail chain. (Examples of retail chain include Ralphs, Seven Eleven, and Ranch 99.) Due to customer privacy, the identity of the chain is hidden, and you only have a numerical ID.
Our sample input is the file 'sales2018.csv.gz'
- sales products
more detailed information about each product in the sales2018.csv.gz dataset.
Our sample input is the file 'sales2018_products.csv.gz'
- distribution cost
A report about the transportation cost for shipping each product between warehouses. Such shipping is needed because certain products are only made in certain warehouses.
Our sample input is the file 'distribution_cost_cleaned.xlsx'
- prices
A list of retail prices of FritoLays product in various price regions, as well as how much FritoLays charges the stores.
Our sample input is the file 'prices_cleaned.csv.gz'

- inventory
A report of the amount of inventory of each product in each warehouse in 2018 sales period 12.
Our sample input is the file '2018_P12_inventory_report.xlsx'
- chain information
Information on the sales channel of each retail chain to customers. (Examples of channels include convenience stores, small grocery, supermarket, etc.) This can be used to classify chains, although certain chains may have multiple sales channels.
Our sample input is the file 'chain_information.csv'
- production capacity
what percentage of production capacity each type of manufacturing process is currently using.
Our sample input is the file 'manufacturing_platform_capacity.csv'

Fritolay.py command line usage

1. CD to the path you saved all the .py file and inputs

Cd \path

```
> cd \Users\h'p\Project
```

2. Input command lines

Then you have to input `ipython Fritolay.py` followed by the names of input files and input variables one by one separated by single space each in the same order shown in the files description above. The codes should not contain any quotation marks.

After inputting the filename, input variables should be added in the following order.

- q:
the cutoff percentage of the sales of core BDCs over total sales decided by Company.
Our sample input is 0.75.
- qm:
the proportion of question mark products in SKU set, decided by Company.
Our sample input is 0.1.

- cc:
the proportion of cash cow products in SKU set, decided by Company.
Our sample input is 0.2.
- st:
the proportion of star products in SKU set, decided by Company.
Our sample input is 0.7.
- h:
the number of healthy products to be included in SKU set, decided by Company.
Our sample input is 10.
- gr:
estimated growth rate on quantities of products, decided by Company.
Our sample input is 0.1.

Sample:

```
C:\Users\h'p\Project>ipython Fritolay.py estimated_margins.csv sales2018.csv.gz sales2018_products.csv.gz distribution_cost_cleaned.xlsx prices_cleaned.csv.gz 2018_P12_inventory_report.xlsx chain_information.csv manufacturing_platform_capacity.csv 0.75 0.1 0.2 0.7 10 0.1 Output_for_fritolay.xlsx
```

```
ipython Fritolay.py estimated_margins.csv sales2018.csv.gz
sales2018_products.csv.gz distribution_cost_cleaned.xlsx
prices_cleaned.csv.gz 2018_P12_inventory_report.xlsx chain_information.csv
manufacturing_platform_capacity.csv 0.75 0.1 0.2 0.7 10 0.1
Output_for_fritolay.xlsx
```

Fritolay.py Jupiter notebook usage

1. Import the function from .py file
from Fritolay import optimize

```
In [1]: from Fritolay import optimize
```

2. Add inputs and run the tool

```
%%time
input_margins='estimated_margins.csv'
input_sales='sales2018.csv.gz'
input_sales_products='sales2018_products.csv.gz'
input_distribution_cost='distribution_cost_cleaned.xlsx'
input_prices='prices_cleaned.csv.gz'
input_inventory='2018_P12_inventory_report.xlsx'
input_chain='chain_information.csv'
input_manufacture="manufacturing_platform_capacity.csv"
q=0.75
qm=0.1
```

```

cc=0.2
st=0.7
h=10
gr=0.1
outputFile='Output_for_fritolay.xlsx'
optimize(input_margins, input_sales, input_sales_products,
          input_distribution_cost, input_prices, input_inventory,
          input_chain, input_manufacture, q, qm, cc, st, h, gr, outputFile)

```

```

In [ ]: %%time
input_margins='estimated_margins.csv'
input_sales='sales2018.csv.gz'
input_sales_products='sales2018_products.csv.gz'
input_distribution_cost='distribution_cost_cleaned.xlsx'
input_prices='prices_cleaned.csv.gz'
input_inventory='2018_P12_inventory_report.xlsx'
input_chain='chain_information.csv'
input_manufacture='manufacturing_platform_capacity.csv'
q=0.75
qm=0.1
cc=0.2
st=0.7
h=10
gr=0.1
outputFile='Output_for_fritolay.xlsx'

optimize(input_margins, input_sales, input_sales_products, \
          input_distribution_cost, input_prices, input_inventory, \
          input_chain, input_manufacture, q, qm, cc, st, h, gr, outputFile)

```

How to interpret results

Output file has two sheets: "Summary" and "Solution"

Summary Solution

First sheet is "Summary". It's the objective value of our optimization tool. It's the total estimated margins minus the total transportation costs.

Sample Output:

Objective Value
151069018.3

Second sheet is "Solution". It's a column of BDC numbers which are chosen to be the SKUs for small format. The following figure shows part of the SKU list.

Sample Output:

BDC
4192011
4192051
28015052
28015816
700306398
3015009
3015071